

GALOIS GROUPS OF SCHUBERT PROBLEMS VIA HOMOTOPY COMPUTATION

ANTON LEYKIN AND FRANK SOTTILE

ABSTRACT. Numerical homotopy continuation of solutions to polynomial equations is the foundation for numerical algebraic geometry, whose development has been driven by applications of mathematics. We use numerical homotopy continuation to investigate the problem in pure mathematics of determining Galois groups in the Schubert calculus. For example, we show by direct computation that the Galois group of the Schubert problem of 3-planes in \mathbb{C}^8 meeting 15 fixed 5-planes non-trivially is the full symmetric group S_{6006} .

INTRODUCTION

Numerical homotopy continuation [26] gives a method to find all solutions to a system of polynomials with finitely many solutions. Current parallel implementations [18] can solve systems with over 40 million solutions [19]. The emerging field of *numerical algebraic geometry* [25, 26] uses numerical homotopy continuation as a foundation for algorithms to study algebraic varieties. While numerical algebraic geometry was developed for applications of mathematics, we apply it in pure mathematics, computing Galois groups of enumerative-geometric problems from the Schubert calculus, called *Schubert problems*.

Along with [2], this is one of the first applications of numerical algebraic geometry to a problem in pure mathematics.

Jordan introduced Galois groups of enumerative problems in 1870 [11] and Harris laid their modern foundations in 1979 [8], showing that the algebraic Galois group is equal to a geometric monodromy group. Byrnes [4, Section 5] used Harris's theory to prove that the general problem of placing poles with static output feedback in linear systems theory was not solvable by radicals. He used numerical homotopy continuation to show that a particular Galois group arising in pole placement was the full symmetric group, S_5 . Underlying this calculation was a Schubert problem. Vakil [32] applied his geometric Littlewood-Richardson rule [31] to study Galois groups of Schubert problems and showed that many Schubert problems have Galois group containing the alternating group.

A Schubert problem is *simple* if it involves no more than two Schubert conditions of codimension more than 1. Simple Schubert problems are natural to study [28, 29] and among all Schubert problems on a given Grassmannian, they have the largest intersection numbers, so they are the most challenging for direct computation. They may also be formulated as complete intersections, which is a restriction imposed by our software.

2000 *Mathematics Subject Classification.* 14N15, 65H20.

Key words and phrases. Polynomial homotopy continuation, Schubert problem, Galois group.

Leykin and Sottile supported by the Institute for Mathematics and its Applications and Sottile by NSF grants CAREER DMS-0538734 and DMS-0701050.

Numerical Theorem. *The Galois group of the Schubert problem of 3-planes in \mathbb{C}^8 meeting 15 fixed 5-planes non-trivially is the full symmetric group S_{6006} .*

This is a numerical theorem, as our software does not certify its output. We have computed Galois groups of scores of other simple Schubert problems, including one on the Grassmannian of 4-planes in \mathbb{C}^8 having 8580 solutions and ones on Grassmannians of 3-planes in \mathbb{C}^8 and in \mathbb{C}^9 having 10329 and 17589 solutions, respectively. In every case, we find that the Galois group is the full symmetric group. Table 2 in Section 3.2 records some of these calculations. Based on this evidence, we conjecture that every simple Schubert problem on a Grassmannian has Galois group equal to the full symmetric group.

Not all Schubert problems have Galois group equal to the full symmetric group. Using an idea of Derksen, Vakil [32] gives some Schubert problems on Grassmannians whose Galois group is not the full symmetric group and Ruffo, et. al. [21] give one on a particularly small flag manifold. None of these examples can be studied with our software, which requires the Schubert problem to have a formulation as a complete intersection.

Our software has two implementations in Maple which use homotopy continuation to compute elements in the Galois groups and either Maple or GAP [6] to determine if these elements generate the full symmetric group. For the continuation, both implementations use PHCpack [33] through its Maple interface PHCmaple [17] and the second may also call Bertini [1]. The advantages of Bertini are that it can use arbitrary precision and it gives an independent verification of our results.

Numerical techniques give insight into some mathematical properties that are far beyond the reach of other methods. For example, Billey and Vakil [3] studied Galois groups of Schubert problems using symbolic methods. The largest problem that they treated (showing its Galois group is the full symmetric group) had 9 solutions on the Grassmannian of 2-planes in \mathbb{C}^6 , and they stated that the Schubert problem on this Grassmannian having 14 solutions was computationally infeasible.

The largest simple Schubert problem which we have solved symbolically has 91 solutions [21, §5.3]. In contrast, numerical methods allow us to solve Schubert problems with as many as 17589 solutions. These examples actually underestimate the gap between the computational possibilities of symbolic and numeric methods, because they were performed on serial machines.

Current and (likely) future increases in computer power will come from multiple core and distributed computing. This is a break with the past, when improvements in computational power came from increasing the clock speed of single-processor units. Symbolic algorithms have limited potential in this regime, as Gröbner basis computation appears to be intrinsically serial and thus can not be efficiently parallelized. In contrast, numerical homotopy continuation is easily parallelized, since its atomic tasks are independent. Thus methods based on numerical continuation will reap the benefits of future parallel architectures. In addition, numerical algorithms typically require less memory than symbolic algorithms. In particular, the sizes of final and intermediate expressions in Gröbner basis computation not only may be large, but also are unpredictable. For these reasons, we feel that the future of computing in algebraic geometry lies in numerical algorithms.

This paper is structured as follows. In Section 1, we describe the basic geometry of Schubert problems and Harris's theory of Galois groups. In Section 2, we explain the use of homotopy continuation for simple Schubert problems. We present our software and algorithms and discuss our results in Section 3, which include the computation described in the Numerical Theorem. We describe future work in Section 4.

1. GALOIS GROUP COMPUTATION OF SCHUBERT PROBLEMS

The Schubert calculus [14] is a method to compute the number of solutions to *Schubert problems*, which are a class of geometric problems involving linear subspaces. The prototypical Schubert problem is the classical problem of four lines: How many lines in space meet four given lines? To answer this, note that three lines ℓ_1, ℓ_2, ℓ_3 lie on a unique doubly-ruled hyperboloid, depicted in Figure 1. These three lines lie in one ruling, while

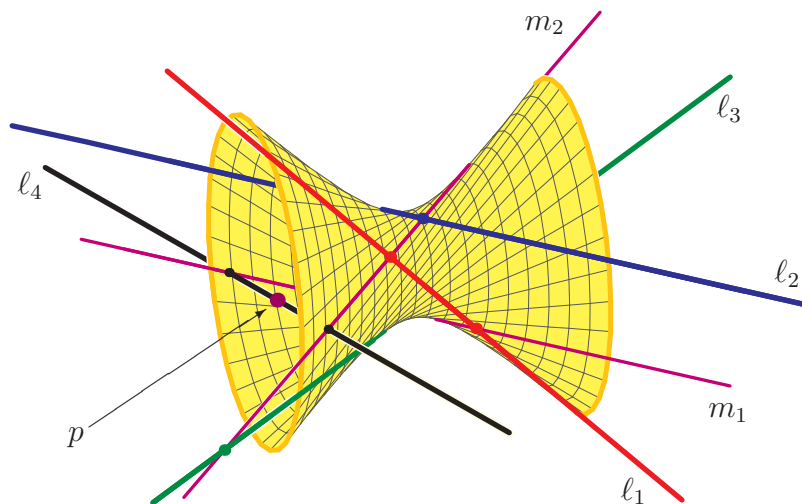


FIGURE 1. The two lines meeting four lines in space.

the second ruling consists of the lines meeting the given three lines. The fourth line ℓ_4 meets the hyperboloid in two points. Through each of these points there is a line in the second ruling, and these are the two lines m_1 and m_2 meeting our four given lines.

The Galois group of this Schubert problem is the group of permutations which are obtained by following the solutions over loops in the space of lines ℓ_1, \dots, ℓ_4 . Rotating ℓ_4 about the point p gives a loop which interchanges the two solution lines m_1 and m_2 , showing that the Galois group is S_2 , the full symmetric group on two letters.

1.1. Schubert problems in the Grassmannian. A typical Schubert problem asks for the linear subspaces of a fixed dimension (*k*-planes) in \mathbb{C}^n that have specified positions (incidence conditions) with respect to some fixed, but otherwise general, linear subspaces. Each incidence condition defines a set of *k*-planes, called a Schubert variety, and the solutions to the Schubert problem are the points of intersection of the corresponding Schubert varieties. We describe this class of problems.

The *Grassmannian* $G(k, n)$ is the set of k -planes in \mathbb{C}^n . This is a complex manifold of dimension $k(n-k)$. The problem of four lines involves the four-dimensional Grassmannian $G(2, 4)$ as a line in (projective) 3-space corresponds to a 2-plane in \mathbb{C}^4 . The set of lines m meeting a fixed line ℓ corresponds to the set of 2-planes M of \mathbb{C}^4 whose intersection with a fixed 2-plane L is at least one-dimensional, and this set of lines m is a Schubert variety. The problem of four lines asks for the points common to four such Schubert varieties, one for each of the given lines $\ell_1—\ell_4$ in projective 3-space.

The specified positions of k -planes in Schubert problems are in reference to flags in \mathbb{C}^n . A *flag* F_\bullet is a sequence of linear subspaces

$$F_\bullet : F_1 \subset F_2 \subset \cdots \subset F_{n-1} \subset F_n = \mathbb{C}^n,$$

where $i = \dim F_i$. The possible positions are encoded by partitions. A *partition* λ is a weakly decreasing sequence of integers

$$\lambda : (n-k) \geq \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k \geq 0.$$

Give a partition λ and a flag F_\bullet , the *Schubert (sub)variety* $Y_\lambda F_\bullet$ of $G(k, n)$ is

$$(1.1) \quad Y_\lambda F_\bullet := \{E \in G(k, n) \mid \dim E \cap F_{n-k+i-\lambda_i} \geq i, \ i = 1, \dots, k\}.$$

This has codimension $|\lambda| := \lambda_1 + \cdots + \lambda_k$ in $G(k, n)$. When $\lambda = \square := (1, 0, \dots, 0)$,

$$Y_\square F_\bullet = \{E \in G(k, n) \mid \dim E \cap F_{n-k} \geq 1\},$$

as the other conditions are redundant. We call \square a *simple Schubert condition* and $Y_\square F_\bullet$ a *simple Schubert variety*. It depends only upon F_{n-k} , so we also write $Y_\square F_{n-k}$. All four Schubert varieties in the problem of four lines are simple.

A *Schubert problem* is a list $(\lambda^1, \dots, \lambda^m)$ of partitions with $|\lambda^1| + \cdots + |\lambda^m| = k(n-k)$. By Kleiman's Transversality Theorem [13], if $F_\bullet^1, \dots, F_\bullet^m$ are general, then the intersection

$$(1.2) \quad Y_{\lambda^1} F_\bullet^1 \cap Y_{\lambda^2} F_\bullet^2 \cap \cdots \cap Y_{\lambda^m} F_\bullet^m$$

is transverse and consists of finitely many k -planes. The number $d(\lambda^1, \dots, \lambda^m)$ of k -planes may be computed using the algorithms in the Schubert calculus (see [14] or [5] or the Introduction to [9]). The problem of four lines is an instance of the Schubert problem $(\square, \square, \square, \square)$ in $G(2, 4)$ and our analysis shows that $d(\square, \square, \square, \square) = 2$.

We study Schubert problems in which all except possibly two Schubert conditions are simple. A *simple Schubert problem* on $G(k, n)$ is one of the form

$$(\lambda, \mu, \underbrace{\square, \dots, \square}_{k(n-k)-|\lambda|-|\mu|}),$$

where λ, μ are not necessarily equal to \square . We speak of the *simple Schubert problem* (λ, μ) on $G(k, n)$ (the $k(n-k) - |\lambda| - |\mu|$ simple conditions \square are understood).

The primary reason for limiting our study to simple Schubert problems in this paper is that these are Schubert problems that are complete intersections, and the off-the-shelf software that we use restricts us to complete intersections.

1.2. Galois groups of Schubert problems. According to Harris [8], Jordan [11] showed how intrinsic structures of some enumerative problems could be understood in terms of Galois theory. Harris took the opposite approach—computing Galois groups of enumerative problems to expose the intrinsic structure of an enumerative problem. He showed that many enumerative problems have Galois group equal to the full symmetric group, demonstrating that these problems had no underlying structures.

Harris’ theory relating Galois groups to monodromy groups begins with a map $f: U \rightarrow V$ of degree d between irreducible complex algebraic varieties U and V . The function field $\mathbb{C}(U)$ of U is a degree d extension of the function field $\mathbb{C}(V)$ of V . These fields may be embedded into the field K of germs of meromorphic functions on a disc around a regular value $v \in V$ of f . If $L \subset K$ is the normalization in K of the extension $\mathbb{C}(U)/\mathbb{C}(V)$, then the Galois group $G = \text{Gal}(L/\mathbb{C}(V))$ acts faithfully on the d points $f^{-1}(v)$, and this gives an embedding $G \hookrightarrow S_d$, where S_d is the symmetric group of the fiber $f^{-1}(v)$.

Replacing U and V by Zariski open subsets if necessary, we may assume that the map $f: U \rightarrow V$ is a degree d covering. A loop in V based at v has d lifts to U , one for each point in the fiber $f^{-1}(v)$. Associating a point in the fiber $f^{-1}(v)$ to the endpoint of the corresponding lift gives a permutation in S_d . This defines the usual permutation action of the fundamental group of V on the fiber $f^{-1}(v)$. The *monodromy group* of the map $f: U \rightarrow V$ is the image of the fundamental group of V in S_d .

Proposition 1.1 (Harris [8]). *For a map $f: U \rightarrow V$ as above, the monodromy group equals the Galois group.*

Given a Schubert problem $(\lambda^1, \dots, \lambda^m)$ on $G(k, n)$, let V be the space of m -tuples $(F_\bullet^1, \dots, F_\bullet^m)$ of flags, a product of flag manifolds. Define U to be the incidence variety

$$(1.3) \quad U := \{(H, F_\bullet^1, \dots, F_\bullet^m) \in G(k, n) \times V \mid H \in Y_{\lambda^i} F_\bullet^i \text{ for } i = 1, \dots, m\}.$$

The fiber of U over a point $H \in G(k, n)$ is a product of the Schubert subvarieties

$$\{F_\bullet^i \mid H \in Y_{\lambda^i} F_\bullet^i\} \quad i = 1, \dots, m$$

of the flag manifold. Each of these is irreducible, and so U is irreducible. Let $f: U \rightarrow V$ be the other projection. Given $v = (F_\bullet^1, \dots, F_\bullet^m) \in V$, the fiber $f^{-1}(v)$ is the intersection (1.2). When v is general, this has $d = d(\lambda^1, \dots, \lambda^m)$ points, so that the map f has degree d . The *Galois group* of the Schubert problem is the Galois group of the extension $\mathbb{C}(U)/\mathbb{C}(V)$. By Proposition 1.1, this is the monodromy group of the map $f: U \rightarrow V$.

The point of this paper is that these monodromy groups may be computed using numerical homotopy continuation. For this, we first compute the points in a single fiber $f^{-1}(v)$. Then, given a loop $\varphi: [0, 1] \rightarrow V$ based at v ($\varphi(0) = \varphi(1) = v$), we numerically follow the points in the fibers $f^{-1}(\varphi(t))$ as t runs from 0 to 1. This computes the lifts of φ and thus the associated monodromy permutation. Computing sufficiently many of these monodromy permutations will enable us to recover the Galois group. While this gives the idea behind our method, we postpone more details until Section 3.1.

2. HOMOTOPY CONTINUATION OF SIMPLE SCHUBERT PROBLEMS

Homotopy continuation is a numerical method to compute all solutions to a system of polynomials given the solutions to a similar system. We use it to find all solutions to a simple Schubert problem and to compute elements of the monodromy group. We first describe the method of numerical homotopy continuation, then discuss polynomial formulations of Schubert problems, and finally explain the Pieri homotopy algorithm [9, 10] to find all solutions to simple Schubert problems.

2.1. Homotopy continuation of polynomial systems. Suppose that we want to find all solutions to a 0-dimensional *target system* of polynomial equations

$$(2.1) \quad f_1(x_1, \dots, x_n) = f_2(x_1, \dots, x_n) = \dots = f_N(x_1, \dots, x_n) = 0,$$

written as $F(x) = 0$. Numerical homotopy continuation finds these solutions if we have a *homotopy*, which is a system $H(x, t)$ of polynomials in $n+1$ variables such that

- (1) The systems $H(x, 1) = 0$ and $F(x) = 0$ both have the same solutions;
- (2) We know all solutions to the *start system* $H(x, 0) = 0$;
- (3) The components of the variety defined by $H(x, t) = 0$ include curves whose projection to \mathbb{C} (via the second coordinate t) is dominant; and
- (4) The solutions to the system $H(x, t) = 0$, where $t \in [0, 1]$, occur at smooth points of curves from (3) in the variety $H(x, t) = 0$.

Given this, we restrict the variety $H(x, t) = 0$ to $t \in [0, 1]$ and obtain finitely many real arcs in $\mathbb{C}^n \times [0, 1]$ which connect (possibly singular) solutions of the target system $H(x, 1) = 0$ to solutions of the start system $H(x, 0) = 0$. We then numerically trace each arc from $t = 0$ to $t = 1$, obtaining all isolated solutions to the target system.

The homotopy is *optimal* if every solution at $t = 0$ is connected to a unique solution at $t = 1$ along an arc. This is illustrated in Figure 2. For simple Schubert problems, the

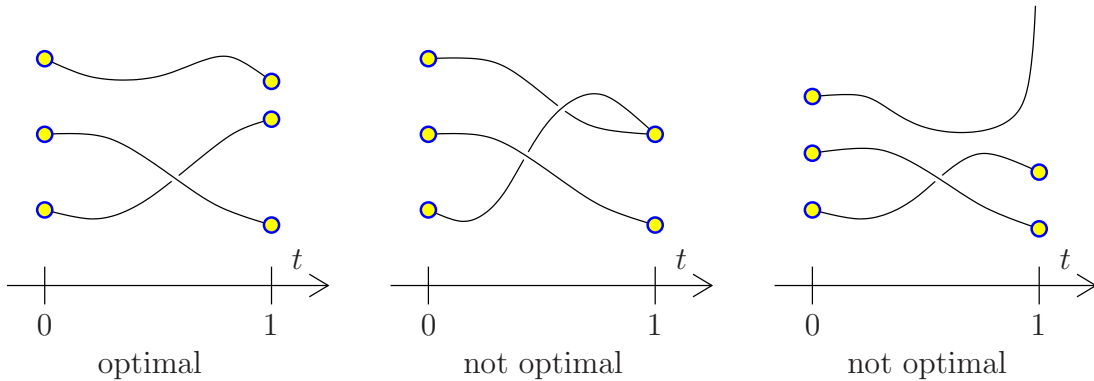


FIGURE 2. Optimal and non-optimal homotopies

Pieri homotopy algorithm is optimal.

Remark 2.1. Homotopy continuation software often constructs a homotopy as follows. Let $F(x)$ be the *target system* (2.1) and suppose we have solutions to a *start system* $G(x)$.

Then for a number $\gamma \in \mathbb{C}$ with $|\gamma| = 1$ define the *linear homotopy*

$$H(x, t) := \gamma t F(x) + (1 - t)G(x).$$

Then $H(x, t)$ satisfies the definition of a homotopy for all but finitely many γ . The software detects the probability 0 event that $H(x, t)$ does not satisfy the definition when it encounters a singularity, and then it recreates the homotopy with a different number γ .

Path following algorithms use predictor-corrector methods, which are conceptually simple for *square systems*, where the number of equations equals the number of variables.

Given a point $(x^{(0)}, t^{(0)})$ on an arc such that $t^{(0)} \in [0, 1)$, the $n \times n$ matrix

$$H_x := \left(\frac{\partial H_i}{\partial x_j} \right)_{i,j=1}^n$$

is regular at $(x^{(0)}, t^{(0)})$, which follows from the definition of the homotopy. Let $H_t := (\partial H_1 / \partial t, \dots, \partial H_n / \partial t)^T$. Given Δt , we set

$$\Delta x := -\Delta t H_x(x^{(0)}, t^{(0)})^{-1} H_t(x^{(0)}, t^{(0)}).$$

For $t^{(1)} = t^{(0)} + \Delta t$, the point $(x', t^{(1)}) = (x^{(0)} + \Delta x, t^{(1)})$ is an approximation to the point $(x^{(1)}, t^{(1)})$ on the same arc. This constitutes a first order predictor step. A corrector step uses the multivariate Newton method for the system $H(x, t^{(1)}) = 0$, refining the approximate solution x' to a solution $x^{(1)}$. In practice, the points $x^{(0)}$ and $x^{(1)}$ are numerical (approximate) solutions, and both the prediction and correction steps require that $\det H_x \neq 0$ at every point where the computation of the Jacobian matrix H_x is done.

When the system is not square, additional strategies must be employed to enable the path following. Fortunately, simple Schubert problems are exactly the class of Schubert problems for which we have an optimal square homotopy (the Pieri homotopy).

Cheater homotopies [20] are optimal homotopies constructed from families of polynomial systems. For example, given a Schubert problem $(\lambda^1, \dots, \lambda^m)$, let V be the space of all m -tuples $(F_\bullet^1, \dots, F_\bullet^m)$ of flags. The total space of the Schubert problem

$$U := \{(H, F_\bullet^1, \dots, F_\bullet^m) \in G(k, n) \times V \mid H \in Y_{\lambda^i} F_\bullet^i \text{ for } i = 1, \dots, m\}$$

is defined by equations (see Section 2.2) depending upon the point $(F_\bullet^1, \dots, F_\bullet^m) \in V$. If $\varphi: \mathbb{C} \rightarrow V$ is an embedding of \mathbb{C} into V in which $\varphi(0)$ and $\varphi(1)$ are general m -tuples of flags and we write $\varphi(t) = (F_\bullet^1(t), \dots, F_\bullet^m(t))$, then

$$\varphi^* U = \{(H, F_\bullet^1(t), \dots, F_\bullet^m(t)) \mid H \in Y_{\lambda^i} F_\bullet^i(t) \text{ for } i = 1, \dots, m\}.$$

This is defined by a system $H(x, t) = 0$, which gives an optimal homotopy. We use this particular cheater homotopy to compute permutations in monodromy groups, called *monodromy permutations*. We give more details in Section 3.1.

There, we describe the Pieri homotopy algorithm, which is a cheater homotopy where only one flag in $\varphi(t)$ actually moves and the others remain fixed. The moving flag is in general position when $t = 1$, but in a particular special position when $t = 0$, so that the Schubert problem becomes a union of other Schubert problems (whose solutions were previously computed and thus are known).

2.2. Equations for Schubert problems. Polynomial homotopy continuation methods require that our geometric problems are modeled by a system of polynomial equations. For efficiency, the number of variables should be minimized. We describe equations for Schubert varieties and then model Schubert problems by systems of equations which minimize the number of variables, stated in Proposition 2.2 below.

Represent a k -plane in \mathbb{C}^n as the row space of a k by n matrix E with full rank and a flag by an invertible n by n matrix F_\bullet of constants, where the i -dimensional subspace in the flag is the row space of the first i rows F_i of the matrix. The condition from (1.1) that $\dim E \cap F_{n-k+i-\lambda_i} \geq i$ is

$$(2.2) \quad \text{rank} \begin{bmatrix} E \\ F_{n-k+i-\lambda_i} \end{bmatrix} \leq n - \lambda_i,$$

which is given by the vanishing of the determinants of all $n+1-\lambda_i$ by $n+1-\lambda_i$ submatrices of the $n+i-\lambda_i$ by n matrix in (2.2). When $\lambda_i = 0$ the condition (2.2) is empty.

Write $\mathcal{E}(E, F_\bullet, \lambda)$ for the system consisting of these $\sum_i \binom{n+i-\lambda_i}{n+1-\lambda_i} \binom{n}{n+1-\lambda_i}$ equations. The codimension $|\lambda|$ equals the number of equations only when $\lambda = \square$. In that case, $\mathcal{E}(E, F_\bullet, \square)$ consists of the single equation

$$(2.3) \quad \det \begin{bmatrix} E \\ F_{n-k} \end{bmatrix} = 0.$$

Since any two flags in general position are conjugate under a linear transformation, we always assume that two flags in (1.2) are fixed. Let the flag F_\bullet be defined by setting F_i to be the span of $e_n, e_{n-1}, \dots, e_{n+1-i}$, where e_1, \dots, e_n form the standard basis for \mathbb{C}^n . The Schubert variety $Y_\lambda F_\bullet$ has an open subset isomorphic to $\mathbb{C}^{k(n-k)-|\lambda|}$ consisting of k -planes that are the row space of an echelon matrix of the form

$$\begin{bmatrix} 0 \cdots 0 & 1 & * \cdots * & 0 & * \cdots * & 0 & * \cdots * \\ 0 \cdots 0 & 0 & 0 \cdots 0 & 1 & * \cdots * & 0 & * \cdots * \\ \vdots & & & \vdots & \ddots & \vdots & \vdots \\ 0 \cdots 0 & 0 & 0 \cdots 0 & 0 & 0 \cdots 0 & 1 & * \cdots * \end{bmatrix}$$

where $1 + \lambda_k, 2 + \lambda_{k-1}, \dots, k + \lambda_1$ are the columns with 1s and $*$ represents some number.

To further reduce the number of variables, let the flag F'_\bullet be defined by setting F'_i to be the span of e_1, \dots, e_i . The *skew Schubert variety* [30] (or Richardson variety),

$$Y_{\lambda, \mu} := Y_\lambda F_\bullet \cap Y_\mu F'_\bullet$$

has an open subset parameterized by matrices of the form

$$E_{\lambda, \mu} := \begin{bmatrix} 0 & 1 & * \cdots * & 0 \cdots 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 \cdots 0 & 1 & * \cdots * & 0 \end{bmatrix}$$

whose entries $a_{i,j}$ are

$$\begin{cases} 1 & \text{if } j = i + \lambda_{k+1-i}, \\ * & \text{if } i + \lambda_{k+1-i} < j \leq n - k + i - \mu_i, \\ 0 & \text{otherwise.} \end{cases}$$

This parameterization is one-to-one if the product of the rightmost entries is non-zero,

$$0 \neq \prod_{i=1}^k a_{i, n-k+i-\mu_i}.$$

On the left below is $E_{\square, \square}$ in $G(2, 4)$ and on the right is $E_{210, 110}$ in $G(3, 7)$.

$$(2.4) \quad \begin{bmatrix} 1 & x & 0 & 0 \\ 0 & 0 & 1 & y \end{bmatrix} \quad \begin{bmatrix} 1 & a & b & c & 0 & 0 & 0 \\ 0 & 0 & 1 & d & e & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & f & g \end{bmatrix}$$

Given a Schubert problem $\lambda, \mu, \nu^1, \dots, \nu^m$, we will always take two of the flags to be these coordinate flags F_\bullet and F'_\bullet , and consider intersections of the form

$$Y_{\lambda, \mu} \cap Y_{\nu^1} F_\bullet^1 \cap \dots \cap Y_{\nu^m} F_\bullet^m,$$

where the flags $F_\bullet^1, \dots, F_\bullet^m$ are general. By Kleiman's transversality theorem, all intersections will lie in the subset of $Y_{\lambda, \mu}$ that is parameterized by matrices from $E_{\lambda, \mu}$, and thus are solutions to the system of equations given by

$$\mathcal{E}(E_{\lambda, \mu}, F_\bullet^1, \nu^1), \dots, \mathcal{E}(E_{\lambda, \mu}, F_\bullet^m, \nu^m).$$

This system is not necessarily square unless it is a simple Schubert problem. Since the homotopy continuation software we use is for square systems of polynomials, we restrict ourselves to simple Schubert problems. Write G for the $n-k$ by n matrix F_{n-k} . Then $\mathcal{E}(E_{\lambda, \mu}, F_\bullet, \square)$ is a single equation (2.3) that depends only on G .

Proposition 2.2. *A simple Schubert problem (λ, μ) on $G(k, n)$ is given by $m := k(n - k) - |\lambda| - |\mu|$ matrices G_1, \dots, G_m each of size $n-k$ by n , and the solutions are modeled by the system of equations*

$$(2.5) \quad \det \begin{bmatrix} E_{\lambda, \mu} \\ G_1 \end{bmatrix} = \det \begin{bmatrix} E_{\lambda, \mu} \\ G_2 \end{bmatrix} = \dots = \det \begin{bmatrix} E_{\lambda, \mu} \\ G_m \end{bmatrix} = 0.$$

For example, the simple Schubert problem \square, \square on $G(2, 4)$ is modeled by

$$(2.6) \quad \det \begin{bmatrix} 1 & x & 0 & 0 \\ 0 & 0 & 1 & y \\ g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \end{bmatrix} = \det \begin{bmatrix} 1 & x & 0 & 0 \\ 0 & 0 & 1 & y \\ g'_{11} & g'_{12} & g'_{13} & g'_{14} \\ g'_{21} & g'_{22} & g'_{23} & g'_{24} \end{bmatrix} = 0,$$

where $G_1 = (g_{ij})$ and $G_2 = (g'_{ij})$ are matrices of constants.

2.3. Pieri homotopy algorithm. We describe the simplified version of the Pieri homotopy algorithm [9, 10] that we use. The Pieri homotopy algorithm finds all solutions to those Schubert problems where all except possibly two partitions consist of a single part, $(a, 0, \dots, 0)$. It is based on subtle geometric degenerations constructed in [27]. Both the algorithm and the degenerations enjoy a dramatic simplification for simple Schubert problems. The degenerations for these simple Schubert problems were introduced by Schubert [22, 23].

Example 2.3. Consider the simple Schubert problem $(\lambda, \mu) = (210, 110)$ in $G(3, 7)$. Begin with local coordinates (2.4) for $E_{210,110}$

$$E := E_{210,110} = \begin{bmatrix} 1 & a & b & c & 0 & 0 & 0 \\ 0 & 0 & 1 & d & e & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & f & g \end{bmatrix}$$

There are four columns, 1, 2, 3, and 6 not of the form $n+i-k-\mu_i$. Let G be a general 4-plane represented by a matrix in which these columns form a identity matrix

$$(2.7) \quad G := \begin{bmatrix} 1 & 0 & 0 & * & * & 0 & * \\ 0 & 1 & 0 & * & * & 0 & * \\ 0 & 0 & 1 & * & * & 0 & * \\ 0 & 0 & 0 & * & * & 1 & * \end{bmatrix}.$$

Let $G(t)$ be this matrix with each entry $*$ scaled by t . Then

$$(2.8) \quad \det \begin{bmatrix} E \\ G(t) \end{bmatrix} = ceg + t(*c + \dots + *cef) + t^2(* + \dots + *bef) + t^3(*f + *af),$$

where each $*$ again represents a fixed number.

When $t = 0$, the expression (2.8) becomes ceg . Let us investigate the consequences of $ceg = 0$. If we set $c = 0$ in $E_{210,110}$, we get $E_{210,210}$ and if we set $g = 0$, we get $E_{210,111}$,

$$E_{210,210} = \begin{bmatrix} 1 & a & b & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & d & e & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & f & g \end{bmatrix} \quad E_{210,111} = \begin{bmatrix} 1 & a & b & c & 0 & 0 & 0 \\ 0 & 0 & 1 & d & e & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & f & 0 \end{bmatrix}.$$

(If $e = 0$, then the row operation $R_1 \leftarrow R_1 - cR_2$ gives a matrix with $c = 0$, which lies in $E_{210,210}$.) This computation in local coordinates shows that

$$Y_{210,110} \cap Y_{\square}G(0) = Y_{210,210} \cup Y_{210,110}.$$

Now suppose that we have general 4 by 7 matrices G_1, \dots, G_6, G_7 , and we wish to solve the instance of the simple Schubert problem $(210, 110)$:

$$(2.9) \quad \det \begin{bmatrix} E \\ G_1 \end{bmatrix} = \dots = \det \begin{bmatrix} E \\ G_6 \end{bmatrix} = \det \begin{bmatrix} E \\ G_7 \end{bmatrix} = 0,$$

where G_7 is the matrix (2.7). Replacing G_7 by $G(t)$ gives a homotopy of 7 equations in the coordinates a, \dots, g where 6 equations are fixed (2.9) and one depends on t (2.8). When $t = 0$, the latter becomes $ceg = 0$ and the system splits into subsystems on $E_{210,210}$ and $E_{210,111}$ involving the matrices G_1, \dots, G_6 . Numerical continuation along this homotopy uses solutions to these smaller problems to obtain solutions to the system (2.9).

Write $\lambda \leq \nu$ if the components of the vector $\nu - \lambda$ are either 0 or 1, with exactly one 1. For example, $110 \leq 210$ and $110 \leq 111$. Given partitions λ, μ , define the $(n-k)$ -plane

$$G_\mu := \{e_i \mid i \notin \{n-k+j-\mu_j, \text{ for } j = 1, \dots, k\}\}.$$

Then if the non-zero entries of the matrix $E_{\lambda,\mu}$ are $a_{i,j}$, we have

$$(2.10) \quad \det \begin{bmatrix} E_{\lambda,\mu} \\ G_\mu \end{bmatrix} = \prod_{i=1}^k a_{i,n-k+i-\mu_i} ,$$

which is the product of the rightmost non-zero entries in the rows of $E_{\lambda,\mu}$. This determinant defines the charts $E_{\lambda,\nu}$ for $\mu < \nu$. A *child problem* for the simple Schubert problem (λ, μ) is one of the form (λ, ν) with $\mu < \nu$.

The Pieri homotopy algorithm finds all solutions to a simple Schubert problem (λ, μ) with fixed (but general) $(n-k)$ -planes G_1, \dots, G_m where $m + |\lambda| + |\mu| = k(n-k)$. We assume that we are given all solutions to all child problems (λ, ν) with $\mu < \nu$ and the $(n-k)$ -planes are G_1, \dots, G_{m-1} . If we let $G_m(t)$ be a 1-parameter family of $(n-k)$ -planes with $G_m(1) = G_m$ and $G_m(0) = G_\mu$, then we obtain a homotopy

$$\det \begin{bmatrix} E_{\lambda,\mu} \\ G_1 \end{bmatrix} = \dots = \det \begin{bmatrix} E_{\lambda,\mu} \\ G_{m-1} \end{bmatrix} = \det \begin{bmatrix} E_{\lambda,\mu} \\ G_m(t) \end{bmatrix} = 0 .$$

When $t = 0$ this is the disjunction of child problems and when $t = 1$, it is the problem we wish to solve.

This method recursively finds solutions to parent problems given solutions to their child problems. The depth of this recursion equals the dimension of the skew Schubert variety corresponding to the simple Schubert problem we wish to solve, i.e., the number of variables in the corresponding equations. The base case of this recursion is when $|\lambda| + |\mu| = k(n-k)$, for then $E_{\lambda,\mu}$ is empty unless $\lambda_i + \mu_{k+1-i} = n-k$ for $i = 1, \dots, k$, and in that case, $E_{\lambda,\mu}$ gives the k -plane spanned by $\{e_{i+\lambda_i} \mid i = 1, \dots, k\}$.

These homotopies are optimal. This is because they only follow solutions to the given Schubert problem and because the number $d(\lambda, \mu)$ of solutions to a simple Schubert problem satisfies the same recursion as the number of paths that are followed. Namely, if $|\lambda| + |\mu| = k(n-k)$ then $d(\lambda, \mu) = 0$ unless $\lambda_i + \mu_{k+1-i} = n-k$ for $i = 1, \dots, k$, and then it equals 1. If $|\lambda| + |\mu| < k(n-k)$, then

$$d(\lambda, \mu) = \sum_{\mu < \nu} d(\lambda, \nu) .$$

Remark 2.4. We do not quite use the homotopy we just described, as the equations involving t will in general have degree in t at least the minimum of k and $n-k$. Instead, we use the convex combination of the equations

$$(2.11) \quad \gamma t \det \begin{bmatrix} E_{\lambda,\mu} \\ G_m \end{bmatrix} + (1-t) \det \begin{bmatrix} E_{\lambda,\mu} \\ G_\mu \end{bmatrix} = 0 .$$

Here, $\gamma \in \mathbb{C}$ has norm 1, $|\gamma| = 1$. This has degree 1 in the homotopy parameter t .

Doing this, the intermediate solutions will not necessarily be solutions to the Schubert problem, so we need to argue that this homotopy will remain optimal. For this, we appeal a little to the geometry of the Grassmannian. The equations we use define hyperplane sections of the Grassmannian in its Plücker embedding, and the number of solutions $d(\lambda, \mu)$ to the Schubert problem turns out to be the degree of the variety $Y_{\lambda,\mu}$. Thus, replacing a family of hyperplanes defined by Schubert conditions (as in the Pieri homotopy) by

an arbitrary pencil of hyperplanes (2.11) will still give an optimal homotopy between solutions to the child problems and solutions to the parent problem.

3. DESCRIPTION OF SOFTWARE

We divide the computation of the Galois group of a Schubert problem into three tasks.

- (1) Compute the solutions of a general instance of the problem, called a *master set*.
- (2) Use cheater homotopies to compute *monodromy permutations* of the master set.
- (3) Determine the *group* generated by the monodromy permutations.

The first task may be accomplished by the brute-force application of a polynomial system solver. This is, however, inefficient. In Table 1, we compare the number of solutions to simple Schubert problems with $\lambda = \mu = \square$ to the number of homotopy paths followed in polyhedral homotopies (as in the black-box solver of PHCpack). This is the volume of the associated Newton polytope and was computed with Polymake [7]. This

k, n	2,6	2,7	2,8	2,9	2,10	2,11	3,6	3,7	3,8	4,6	4,7	5,7
solutions	14	42	132	429	1430	4862	42	462	6006	14	462	42
paths	18	67	248	919	3426	12843	130	3004	74645	42	7156	364

TABLE 1. Inefficiency of polyhedral homotopy for simple Schubert problems

shows that the Pieri homotopy algorithm is an efficient alternative.

Subsection 3.1 discusses the second task.

The theory for the third task is beyond the scope of this paper. Based on preliminary computations, we conjectured that the Galois group is always the full symmetric group, and therefore we only check this. The fastest routine we have found to accomplish this is the `isNaturalSymmetricGroup` function of GAP [6].

In Section 3.2, we discuss implementations of our algorithm. Implementations and documentation of our computations are available at our website [16].

3.1. Computing monodromy permutations. Suppose that we have a master set of solutions to the simple Schubert problem (λ, μ) on $G(k, n)$, which is modeled by the system of equations

$$(3.1) \quad \det \begin{bmatrix} E_{\lambda, \mu} \\ G_1 \end{bmatrix} = \det \begin{bmatrix} E_{\lambda, \mu} \\ G_2 \end{bmatrix} = \cdots = \det \begin{bmatrix} E_{\lambda, \mu} \\ G_m \end{bmatrix} = 0,$$

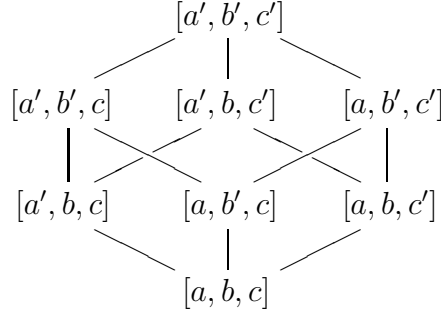
where G_1, \dots, G_m are fixed $(n-k)$ -planes. We follow these solutions along loops in the space of m -tuples of $(n-k)$ -planes to compute monodromy permutations.

The off-the-shelf homotopy continuation software we use requires that the equations are linear in the homotopy parameter t , and so we follow piece-wise linear loops. For these, we fix all of the G_i except G_m , and exploit the linearity of the determinant in the row vectors g_1, \dots, g_{n-k} of G_m . If g'_i is a vector not in G_m and we replace the row g_i by the convex combination $(1-t)g_i + tg'_i$, obtaining the pencil of planes $G_m(t)$, then

$$\begin{aligned} \det \begin{bmatrix} E_{\lambda, \mu} \\ G_m(t) \end{bmatrix} &= (1-t) \det \begin{bmatrix} E_{\lambda, \mu} \\ G_m \end{bmatrix} + t \det \begin{bmatrix} E_{\lambda, \mu} \\ G_m(1) \end{bmatrix} \\ &= (1-t)F(g_1, \dots, g_m) + tF(g_1, \dots, g'_i, \dots, g_m). \end{aligned}$$

Replacing the equation in (3.1) involving G_m with this equation gives a linear homotopy between the system (3.1) and one with $G_m(1)$ in place of G_m .

Given a different $(n-k)$ -plane G'_m spanned by g'_1, \dots, g'_{n-k} , we use these vectors to generate loops along which we can compute monodromy permutations. Suppose for illustration that $n-k = 3$ and the vectors are $[a, b, c]$ for G_m and $[a', b', c']$ for G'_m . The different pencils that we may create correspond to the edges of a cube.



Our software offers 3 strategies to generate loops.

- **long loop** goes from a vertex of the cube to its opposite and back,

$$\begin{aligned} [a, b, c] &\rightarrow [a', b, c] \rightarrow [a', b', c] \rightarrow [a', b', c'] \rightarrow \\ &\rightarrow [a, b', c'] \rightarrow [a, b, c'] \rightarrow [a, b, c]. \end{aligned}$$

- **short loop** uses only a square,

$$[a, b, c] \rightarrow [a', b, c] \rightarrow [a', b', c] \rightarrow [a, b', c] \rightarrow [a, b, c].$$

- **half loop** makes use of just one edge,

$$[a, b, c] \xrightarrow{1} [a', b, c] \xrightarrow{\gamma} [a, b, c],$$

where the second homotopy is modified via a random number $\gamma \in \mathbb{C}$,

$$H_\gamma := (1-t)F(a', b, c) + \gamma t F(a, b, c).$$

While we do not offer a proof that these loops will suffice to find all non-trivial permutations, we remark that they do suffice in the examples we considered.

Example 3.1. Suppose that we have the simple Schubert problem (\square, \square) on $G(2, 4)$ as given by (2.6) with

$$\begin{aligned} G_1 &= \begin{bmatrix} -55 - 8i & 17 + 15i & 40 + 99i & -17 - 38i \\ -67 + 25i & -82 - 55i & -99 - 80i & -21 - 85i \end{bmatrix} \\ G_2 &= \begin{bmatrix} 66 + 53i & -73 - 14i & 85 + 5i & 67 + 16i \\ -53 - 85i & 36 - 25i & 2 + 81i & -58 + 35i \end{bmatrix}. \end{aligned}$$

Its solutions m_1 and m_2 are

$$\begin{aligned} m_1 &:= \begin{bmatrix} 1 & -0.23714 - .0028980i & 0 & 0 \\ 0 & 0 & 1 & -.51680 - .10520i \end{bmatrix} \\ m_2 &:= \begin{bmatrix} 1 & .97009 + 1.2705i & 0 & 0 \\ 0 & 0 & 1 & .44336 + .38248i \end{bmatrix} \end{aligned}$$

The short loop strategy with

$$G' = \begin{bmatrix} 33 - 84i & 21 - i & 59 + 94i & -94 + 89i \\ -15 - 19i & 29i & 79 + 51i & 89 + 3i \end{bmatrix}$$

creates a non-trivial monodromy permutation. The paths followed during the homotopies are drawn in Figure 3, where the large circles are the values taken at the endpoints of the different homotopies.

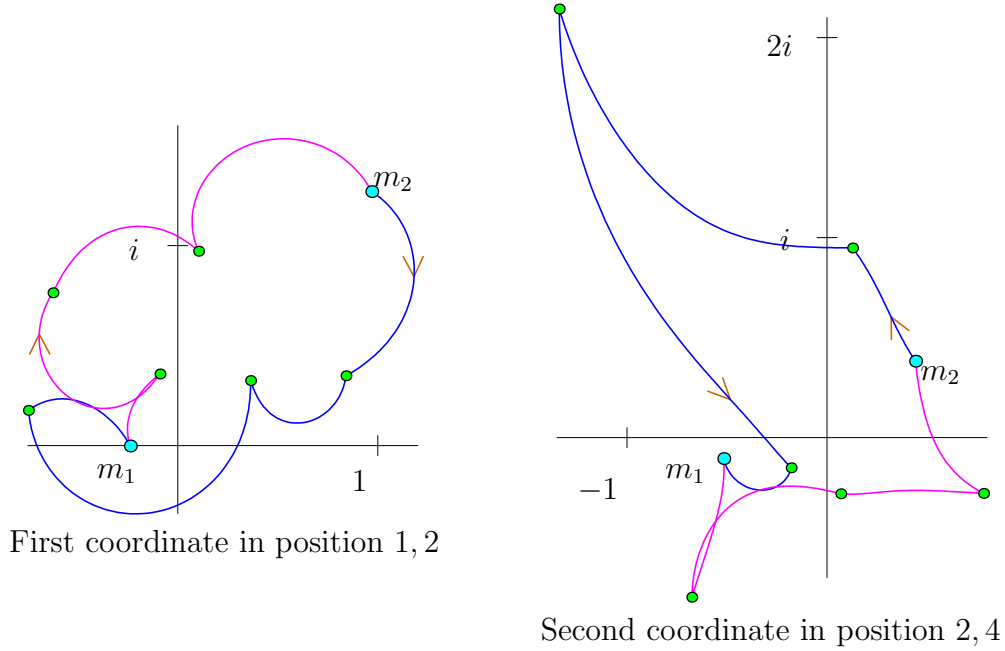


FIGURE 3. Paths tracked in Example 3.1.

3.2. Implementation. We have two Maple implementations of our algorithms using the package PHCmaple [17] to interface with PHCpack [33], which performs the numerical polynomial homotopy continuation. PHCmaple produced the graphic of Figure 3. The second implementation may alternatively call Bertini [1].

Our prototype implementation was carried out entirely in Maple to take advantage of Maple packages to generate the equations and to manage the monodromy group, while using the black-box solver in PHCpack to compute the master sets of solutions. The largest problem this implementation could treat was the simple Schubert problem $(210, 200)$ on $G(3, 7)$ —it showed that the Galois group is the full symmetric group S_{91} . Previously, the largest Schubert problem whose Galois group that was proven to be the full symmetric group was the simple Schubert problem $(20, 10)$ on $G(2, 6)$ with 9 solutions [3].

Our second implementation also uses Maple and either PHCpack or Bertini. However, it relies on the Pieri homotopy algorithm to compute master sets of solutions and GAP to manage the monodromy groups, removing the two main computational bottlenecks of the prototype. The largest problems this implementation has treated are $(2100, \square)$ in $G(4, 8)$ with 8580 solutions, $(210, 210)$ in $G(3, 8)$ with 10329 solutions and $(210, 200)$

on $G(3, 9)$ with 17589 solutions. It showed that all of these have Galois group equal to the full symmetric group. The computation for the problem (\square, \square) on $G(3, 8)$ with 6006 solutions is the basis of the Numerical Theorem.

We computed Galois groups of the simple Schubert problems (\square, \square) on all small Grassmannians, using the short loops strategy. They were run on several different computers, including an AMD Athlon 64 Dual Core Processor 4600+ with CPU clock speed of 2400 MHz and 1 GB of memory whose timings (using PHCpack) are reported in Table 2. These

k, n	2,4	2,5	2,6	2,7	2,8	2,9	2,10
solutions	2	5	14	42	132	429	1430
time	12s	27s	19s	51s	4.2m	20.5m	2.6h
permutations	4	6	5	6	7	4	7

k, n	3,5	3,6	3,7	3,8	3,9	4,6	4,7	4,8
solutions	5	42	462	6006	17589	14	462	8580
time	12s	35s	17.9m	18.6h	78.2h	15s	23.5m	44.5h
permutations	4	4	5	6	7	5	5	7

s := seconds, m := minutes, h := hours

TABLE 2. Timings of Galois group computation

reported times are not CPU times, but actual elapsed (wall clock) time, and so may exceed CPU time by 10 to 20 %. We also record the number of permutations we needed to compute. The entry in $G(3, 9)$ is the Schubert problem (210, 200) and the entry in $G(4, 8)$ is the Schubert problem (2100, \square).

We ran some of these computations in Bertini. It was unable to compute examples in more than 10 variables, and was markedly slower for the largest computations it completed, on $G(2, 8)$, $G(3, 7)$, and $G(4, 7)$. On the other hand, Bertini provided an independent verification that the Galois groups were indeed the full symmetric groups. We remark that Bertini is new software and its efficiency will likely improve.

4. CONCLUSIONS AND FUTURE WORK

This paper demonstrates the feasibility of homotopy continuation as tool to study the Galois groups of enumerative problems. It also implicitly provides several challenges to the numerical homotopy community. Perhaps the most serious is the current lack of certifiability of computations in numerical homotopy software. While numerical methods will increasingly outperform symbolic algorithms in algebraic geometry, they are currently inferior in that their results do not come with certification. Certificates for numerical computations do exist in theory, for example in Shub and Smale's [24] alpha-theory, and there is a need for their implementation. In fact, even more reliable numerical techniques, such as the interval step control proposed in [12], are sidestepped by homotopy continuation software developers mostly due to the perceived complexity of implementation and the expected slower performance in comparison with heuristic methods. Robust, off-the-shelf software to handle polynomial systems that are not complete intersections is also needed to deal with ideals in algebraic geometry, which are typically not complete intersections.

Two further theoretical problems are not addressed in this paper. While the sampling of the fundamental group of the base space provided by the loop-generating heuristics of subsection 3.1 is sufficient to generate the Galois group in the examples we considered, a better understanding of the topology of a complement of an algebraic variety in a Grassmannian is needed to prove that this sampling is always sufficient. Second, we do not know how to certify that the computed set of permutations generates the whole Galois group, if it is not the full symmetric group.

The tools that we use could be applied more systematically to other problems in enumerative geometry. To this end, we plan a comprehensive project exploring the limit of computability of Galois groups of Schubert problems along several fronts. This will involve the software and algorithms described here—perhaps also incorporating HOM4PS [15]. We will also write parallel software implementing algorithms to compute Galois groups of Schubert problems that are not compete intersections, as well as pushing the limits of the symbolic methods of Billey and Vakil [3] and of Vakil’s combinatorial algorithm [32]. This will be a large and distributed computation as in [21] which should give a catalog of the Galois groups of several tens of thousands of Schubert problems.

REFERENCES

- [1] Daniel J. Bates, Jonathan D. Hauenstein, Andrew J. Sommese, and Charles W. Wampler, *Bertini: Software for numerical algebraic geometry*, Available at <http://www.nd.edu/~sommese/bertini>.
- [2] Daniel J. Bates, Chris Peterson, Andrew J. Sommese, and Charles W. Wampler, *Numerical computation of the genus of an irreducible curve within an algebraic set*, Preprint, 2007.
- [3] S. Billey and R. Vakil, *Intersections of Schubert varieties and other permutation array schemes*, Algorithms in Algebraic Geometry (A. Dickenstein, F.-O. Schreyer, and A. J. Sommese, eds.), IMA Volumes in Mathematics and its Applications, vol. 146, Springer New York, 2007, pp. 21–54.
- [4] C. I. Byrnes, *Pole assignment by output feedback*, Three Decades of Mathematical Systems Theory (H. Nijmeijer and J. M. Schumacher, eds.), Lecture Notes in Control and Inform. Sci., vol. 135, Springer-Verlag, Berlin, 1989, pp. 31–78.
- [5] Wm. Fulton, *Young tableaux*, Cambridge University Press, Cambridge, 1997, With applications to representation theory and geometry.
- [6] The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.4.9*, 2006.
- [7] Evgenij Gawrilow and Michael Joswig, *polymake: a framework for analyzing convex polytopes*, Polytopes—combinatorics and computation (Oberwolfach, 1997), DMV Sem., vol. 29, Birkhäuser, Basel, 2000, pp. 43–73.
- [8] J. Harris, *Galois groups of enumerative problems*, Duke Math. J. **46** (1979), 685–724.
- [9] B. Huber, F. Sottile, and B. Sturmfels, *Numerical Schubert calculus*, J. Symb. Comp. **26** (1998), no. 6, 767–788.
- [10] B. Huber and J. Verschelde, *Pieri homotopies for problems in enumerative geometry applied to pole placement in linear systems control*, SIAM J. Control Optim. **38** (2000), no. 4, 1265–1287 (electronic).
- [11] C. Jordan (ed.), *Traité des substitutions*, Gauthier-Villars, Paris, 1870.
- [12] R. Baker Kearfott and Zhaoyun Xing, *An interval step control for continuation methods*, SIAM J. Numer. Anal. **31** (1994), no. 3, 892–914.
- [13] S. Kleiman, *The transversality of a general translate*, Compositio Math. **28** (1974), 287–297.
- [14] S. Kleiman and D. Laksov, *Schubert calculus*, Amer. Math. Monthly **79** (1972), 1061–1082.
- [15] T. Lee, T.Y. Li, and C. Tsai, *Hom4ps-2.0: A software package for solving polynomial systems by the polyhedral homotopy continuation method*, Available at <http://www.math.msu.edu/~li/Software.htm>, 2007.

- [16] A. Leykin and F. Sottile, *Galois groups of Schubert problems*, 2007, www.math.tamu.edu/~sottile/stories/Galois.
- [17] A. Leykin and J. Verschelde, *Interfacing with the numerical homotopy algorithms in PHCpack*, Proceedings of ICMS 2006 (Nobuki Takayama and Andres Iglesias, eds.), 2006, pp. 354–360.
- [18] A. Leykin, J. Verschelde, and Z. Yang, *Parallel homotopy algorithms to solve polynomial systems*, Proceedings of ICMS 2006 (Nobuki Takayama and Andres Iglesias, eds.), 2006, pp. 225–234.
- [19] T.-Y. Li, personal communication.
- [20] T. Y. Li, Tim Sauer, and J. A. Yorke, *The cheater’s homotopy: an efficient procedure for solving systems of polynomial equations*, SIAM J. Numer. Anal. **26** (1989), no. 5, 1241–1251.
- [21] J. Ruffo, Y. Sivan, E. Soprunova, and F. Sottile, *Experimentation and conjectures in the real Schubert calculus for flag manifolds*, Experiment. Math. **15** (2006), no. 2, 199–221.
- [22] H. Schubert, *Anzahl-Bestimmungen für lineare Räume beliebiger Dimension*, Acta. Math. **8** (1886), 97–118.
- [23] ———, *Losung des Charakteritiken-Problems für lineare Räume beliebiger Dimension*, Mittheil. Math. Ges. Hamburg (1886), 135–155, (dated 1885).
- [24] M. Shub and S. Smale, *Complexity of Bézout’s Theorem I: Geometric Aspects*, J. the American Mathematical Society **6** (1993), no. 2, 459–501.
- [25] A. Sommese, J. Verschelde, and C. Wampler, *Introduction to numerical algebraic geometry*, Graduate School on Systems of Polynomial Equations: From Algebraic Geometry to Industrial Applications. 14–25 July 2003, Buenos Aires, Argentina (A. Dickenstein and I. Emiris, eds.), INRIA, 2003, A revised collection of the course notes is scheduled to be published by Springer-Verlag, pp. 229–247.
- [26] A. Sommese and C. Wampler, *The numerical solution of systems of polynomials*, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2005, Arising in engineering and science.
- [27] F. Sottile, *Pieri’s formula via explicit rational equivalence*, Canad. J. Math. **49** (1997), no. 6, 1281–1298.
- [28] ———, *Some real and unreal enumerative geometry for flag manifolds*, Michigan Math. J. **48** (2000), 573–592, Dedicated to William Fulton on the occasion of his 60th birthday.
- [29] ———, *Elementary transversality in the Schubert calculus in any characteristic*, Michigan Math. J. **51** (2003), no. 3, 651–666.
- [30] Richard P. Stanley, *Some combinatorial aspects of the Schubert calculus*, Combinatoire et représentation du groupe symétrique (Actes Table Ronde CNRS, Univ. Louis-Pasteur Strasbourg, Strasbourg, 1976), Springer, Berlin, 1977, pp. 217–251. Lecture Notes in Math., Vol. 579.
- [31] R. Vakil, *A geometric Littlewood-Richardson rule*, Ann. of Math. (2) **164** (2006), no. 2, 371–421, Appendix A written with A. Knutson.
- [32] ———, *Schubert induction*, Ann. of Math. (2) **164** (2006), no. 2, 489–512.
- [33] J. Verschelde, *Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation*, ACM Trans. Math. Softw. **25** (1999), no. 2, 251–276, Software available at <http://www.math.uic.edu/~jan>.

INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS, UNIVERSITY OF MINNESOTA, MINNEAPOLIS, MN 55455, USA

E-mail address: leykin@ima.umn.edu

URL: <http://www.ima.umn.edu/~leykin/>

DEPARTMENT OF MATHEMATICS, TEXAS A&M UNIVERSITY, COLLEGE STATION, TEXAS 77843, USA

E-mail address: sottile@math.tamu.edu

URL: <http://www.math.tamu.edu/~sottile/>